WHAT IS CLAIMED IS:

1. A method for recovering 3D scene structure and camera motion from image data obtained from a multi-image sequence, wherein a reference image of the sequence is taken by a camera at a reference perspective and one or more successive images of the sequence are taken at one or more successive different perspectives by translating and/or rotating the camera, the method comprising the steps of:

(a) determining image data shifts for each successive image with respect to the reference image; the shifts being derived from the camera translation and/or rotation from the reference perspective to the successive different perspectives;

(b) constructing a shift data matrix that incorporates the image data shifts for each image;

(c) calculating a rank-1 factorization from the shift data matrix using SVD, with one of the rank-1 factors being a vector corresponding to the 3D structure and the other rank-1 factor being a vector corresponding to the size of the camera motions;

(d) dividing the successive images into smoothing windows;

(e) recovering the direction of camera motion from the first vector corresponding to the 3D structure by solving a linear equation; and

(f) recovering the 3D structure by solving a linear equation using the recovered camera motion.

2. The method of claim 1, wherein, step (e) includes:

computing a first projection matrix;

recovering camera rotation vectors from the shift data matrix, and the first projection matrix;

computing a second projection matrix; and

recovering the direction of camera translation using the shift data matrix, the reference image, the second projection matrix and the recovered camera rotation vectors.

3. The method of claim 2, wherein step (f) includes recovering the 3D structure from the shift data matrix, the reference image, the recovered camera rotation vectors and the recovered direction of translation vectors.

4. The method of claim 1, further including preliminary steps of:

recovering the rotations of the camera between each successive image; and

warping all images in the sequence toward the reference image, while neglecting the translations.

5. The method of claim 1, wherein step (b) comprises:

computing H and Δ_{QH}, where H is a $(N_p - 3) \times N_p$ matrix defined so that HH^T is the identity matrix and H annihilates the three vectors Ψ_x, Ψ_y, Ψ_z where the three vectors are computed from the reference image as

$\Psi_x \equiv \{\nabla I \cdot \mathrm{r}^{(1)}(\mathrm{p})\}, \Psi_y \equiv \{\nabla I \cdot \mathrm{r}^{(2)}(\mathrm{p})\}, \Psi_z \equiv \{\nabla I \cdot \mathrm{r}^{(3)}(\mathrm{p})$ where

$r^{(1)}(x,y), r^{(2)}(x,y), r^{(3)}(x,y)$ are defined by $\left[\mathrm{r}^{(1)}, \mathrm{r}^{(2)}, \mathrm{r}^{(3)}\right] \equiv \left[\begin{pmatrix} -xy \\ -(1+y^2) \end{pmatrix}, \begin{pmatrix} 1+x^2 \\ xy \end{pmatrix}, \begin{pmatrix} -y \\ x \end{pmatrix}\right]$

and. Δ is a shift data matrix that gives the difference in intensities between each successive image and the reference image and is a $(N_I - 1) \times N_p$ matrix with entries ΔI_n^i ,where ΔI_n^i is the change in (smoothed) intensity with respect to the reference image, and with no smoothing $\Delta I_n^i = I_n^i - I_n^0$, where N_1 is the number of images, N_p is the number of pixels, and where I^i denotes the i-th image, with i=0.1…,$N_I - 1$, and where $I_n^i = I^i(p_n)$ denotes the image intensity at the n-th pixel position in I^i, where I^0 is the reference image, where x and y are the image coordinates of the pixel position and p= (x,y) and where $\Delta_{CH} \equiv C^{-1/2}\Delta \mathrm{H}^T$ where C is a constant matrix with and where the notation $\{V\}$ used to denote a vector with elements given by the V^a .

6. The method of claim 1, wherein step (c) comprises:

computing a rank-1 factorization of $-\Delta_{CH} \approx \mathrm{M}^{(1)}\mathrm{S}^{(1)T}$ where $\mathrm{M}^{(1)}, \mathrm{S}^{(1)T}$ are vectors corresponding to the motion and structure respectively.

7. The method of claim 1, wherein step (c) comprises:

computing a rank-3 factorization of $-\Delta_{CH} \approx \sum_{a=1}^{3} \mathrm{M}^{(a)}\mathrm{S}^{(a)T}$ where $\mathrm{M}^{(a)}, \mathrm{S}^{(a)T}$ are vectors corresponding to the motion and structure respectively;

setting Z_n^{-1} as constant within each window, where Z is the depth from the camera to a 3D scene along the cameras optical axis;

listing each of the pixels so that those in the *k*-th smoothing window have sequential indices n_k, (n_k+1), . . . $(n_{k+1} - 1)$;

computing a first projection matrix by computing a Np x Np projection matrix P_Ω which is block diagonal with zero entries between different smoothing windows, and which annihilates the vectors $\{\nabla I \cdot p\}$, $\{I_x\}$ and $\{I_y\}$ where $\{\nabla I\}$ is a vector containing the gradient of the intensity at each pixel, and I_x and I_y are the gradients of the intensity in the reference image in the *x* and *y* directions;

recovering the three camera rotation vectors includes solving the following equations

$P_\Omega\left(H^T S^{(a)} - \Psi w^{(a)}\right) = 0$ for the 3-vector $w^{(a)}$;

computing a second projection matrix includes computing a $N_p \times N_p$ projection matrix $P_{\hat{T}}^{(a)}$, which is block diagonal with zero entries between different smoothing windows and annihilates $\left(H^T S^{(a)}\right) - \Psi w^{(a)}$ where $w^{(a)}$ is the vector recovered previously;

recovering the directions of camera translations by solving for the directions of translation $\hat{T}^{(a)}$ via

$P_{\hat{T}}^{(a)}\left(-\hat{T}_x^{(a)}\{I_x\} - -\hat{T}_y^{(a)}\{I_y\} + \hat{T}_z^{(a)}\{p \cdot \nabla I\}\right) = 0$ and;

recovering Z_n via

$\left(H^T S^{(a)}\right)_n - \left[\Psi w^{(a)}\right]_n = \tau^{(a)}\left(Z_n^{-1}\left(\hat{T}_z^{(a)} p_n - \left[\hat{T}^{(a)}\right]_2\right) \cdot \nabla I_n\right)$ where $\left[\hat{T}^{(a)}\right]_2$ represents $\hat{T}_x^{(a)}\hat{T}_y^{(a)}$ the x and y component of the translation direction, and where $\tau^{(a)}$ are constants.

8. The method of claim 1, wherein step (d) comprises:

setting Z_n^{-1} as constant within each window, where Z is the depth from the camera to a 3D scene along the cameras optical axis;

listing each of the pixels so that those in the k-th smoothing window have sequential indices n_k, (n_k+1), . . ., $(n_{k+1} - 1)$.

9. The method of claim 2, wherein the step of computing a first projection matrix includes computing a Np x Np projection matrix P_Ω which is block diagonal with zero entries between different smoothing windows, and which annihilates the vectors $\{\nabla I \cdot \mathrm{p}\}, \{I_x\}$ and $\{I_y\}$ where $\{\nabla I\}$ is a vector containing the gradient of the intensity at each pixel, and I_x and I_y are the gradients of the intensity in the reference image in the x and y directions.

10. The method of claim 2, wherein the step of recovering camera rotation vectors includes solving the following equation

$P_\Omega \left(\mathrm{H}^T \mathrm{S}^{(1)} - \Psi w\right) = 0$ for the 3-vector w.

11. The method of claim 9 wherein, the step of computing a second projection matrix includes computing a $N_p \times N_p$ projection matrix P_T, which is block diagonal with zero entries between different smoothing windows and annihilates $\left(H^T S^{(1)}\right) - \Psi \mathrm{w}$ where w is the vector recovered previously.

12. The method of claim 2 wherein, the step of recovering the direction of camera translation includes solving for the direction of translation $\hat{\mathrm{T}}$ via

$$\mathrm{P}_{\hat{T}}\left(-\hat{T}_x\{I_x\}--\hat{T}_y\{I_y\}+\hat{T}_z\{\mathrm{p}\cdot\nabla I\}\right)=0.$$

13. The method of claim 3 wherein, step (f) includes, recovering Z_n via $\left(H^T S^{(1)}\right)_n-[\Psi w]_n=Z_n^{-1}\left(\hat{T}_z\mathrm{p}_n-\left[\hat{\mathrm{T}}\right]_2\right)\cdot\nabla I_n$ where $\left[\hat{\mathrm{T}}\right]_2$ represents $\hat{\mathrm{T}}_x\hat{\mathrm{T}}_y$ the x and y component of the translation direction.

add A4

add B1